

Apa itu ?

API dan Alur Kerja AI

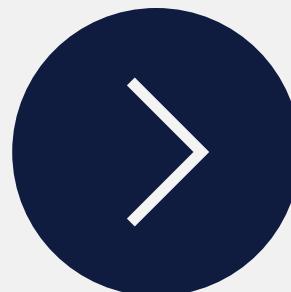
- **API (Application Programming Interface)** adalah antarmuka yang memungkinkan aplikasi atau sistem yang berbeda berinteraksi dan saling bertukar data.
- **Dalam Vibe Coding**, API digunakan untuk menghubungkan AI dengan aplikasi web atau repositori kode, memungkinkan otomatisasi dan pengelolaan kode secara efisien.

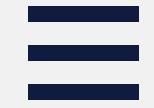
Mengapa Menggunakan API dalam Alur Kerja AI?

- API memungkinkan pengembang untuk menghubungkan aplikasi web dengan AI, mengotomatiskan proses seperti pengambilan data, pengolahan permintaan pengguna, atau pembaruan kode secara langsung.
- Dengan API, pengembang dapat menyinkronkan Cline dengan repositori kode seperti GitHub, yang memungkinkan otomatisasi pengelolaan versi dan penerapan fungsi API dalam proyek.

Mengapa Menggunakan API dalam Alur Kerja AI?

- API memungkinkan pengembang untuk menghubungkan aplikasi web dengan AI, mengotomatiskan proses seperti pengambilan data, pengolahan permintaan pengguna, atau pembaruan kode secara langsung.
- Dengan API, pengembang dapat menyinkronkan Cline dengan repositori kode seperti GitHub, yang memungkinkan otomatisasi pengelolaan versi dan penerapan fungsi API dalam proyek.





Practice

Contoh Praktik dengan Alur Kerja AI dan API:

Instruksi kepada Cline:

Pengembang menulis: "Buat aplikasi web dengan halaman login yang terhubung ke API untuk otentikasi pengguna dan hasilkan suara dengan ElevenLabs."

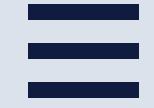


- **Kode yang Dihasilkan:**

- Frontend (React): Halaman login dengan form untuk email dan password.
- Backend (Node.js): API untuk memverifikasi kredensial pengguna dan memberikan sesi login.
- Integrasi API:
 - OpenAI digunakan untuk menganalisis input pengguna (misalnya, memberikan saran atau analisis otomatis berdasarkan permintaan).
 - ElevenLabs digunakan untuk menghasilkan suara dari teks yang diberikan oleh OpenAI. Suara ini bisa digunakan sebagai pengumuman atau interaksi suara dalam aplikasi.

- **Pengujian dan Penyempurnaan:**

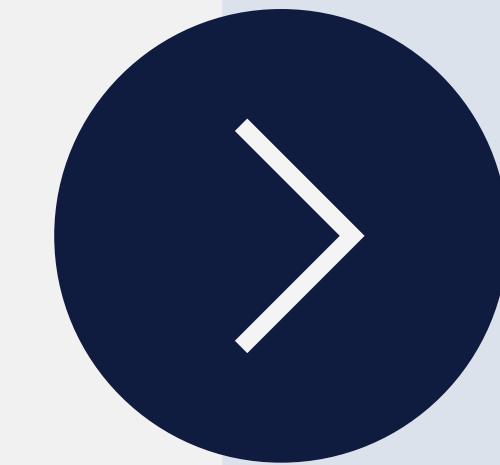
- Pengembang dapat menguji aplikasi dengan menggunakan API untuk memeriksa apakah otentikasi pengguna bekerja dengan baik dan apakah suara pengumuman berhasil dihasilkan dari teks yang diberikan.



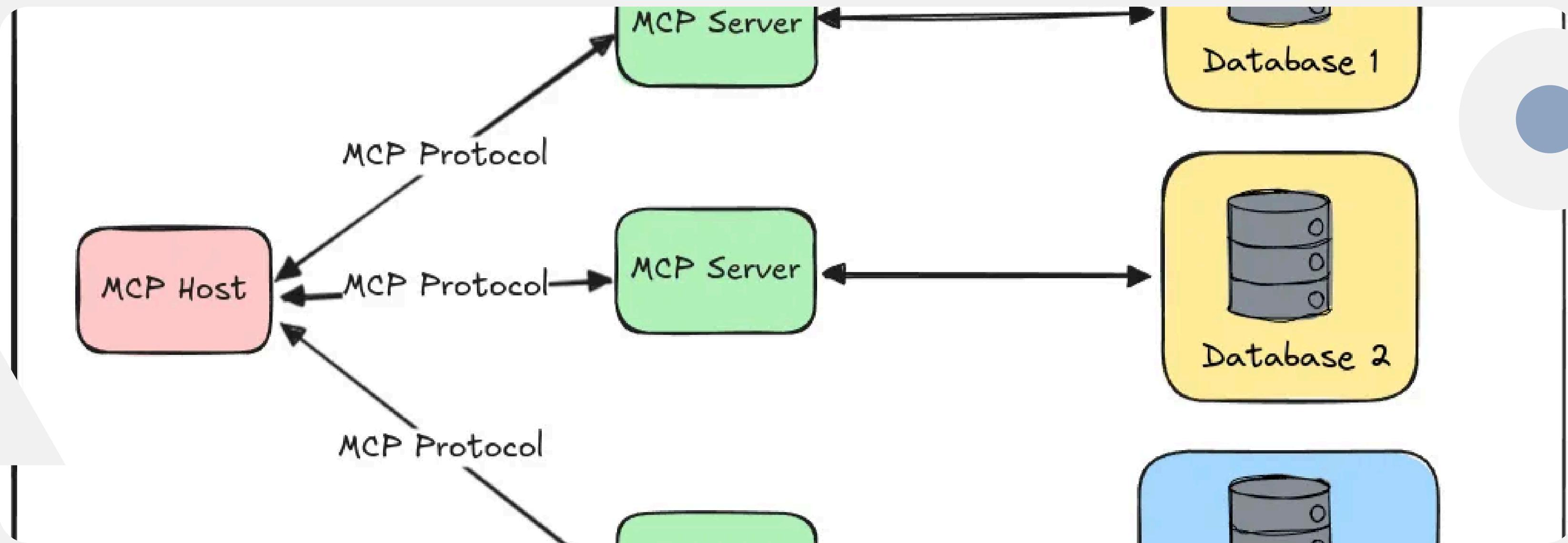
Materi MCP dan Integrasi AI

Day 3

MCP AI



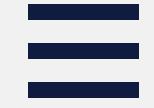
MCP (Message Control Protocol) konsep penting yang dapat berhubungan dalam pengembangan sistem berbasis AI dan pengkodean komunikasi dalam aplikasi atau platform



(Model Context Protocol)

MCP

MCP (Model Context Protocol) adalah sistem plug-and-play yang distandarisasi untuk semua model AI, baik itu GPT-4, Claude, Mistral, atau model AI terbaru yang akan datang. Tujuannya adalah untuk menghubungkan berbagai sumber data, alat, dan API yang digunakan oleh model AI. Dengan menggunakan MCP, Anda dapat mengintegrasikan berbagai data dari database, kode aplikasi, file lokal, hingga API eksternal tanpa perlu membangun integrasi kustom untuk setiap alat yang digunakan.



Integrasi pada Fase LLM



Fase 1: LLMs (Large Language Models)

- Pada awalnya, teknologi seperti ChatGPT dan Claude hanya berfokus pada memprediksi kata berikutnya berdasarkan data pelatihan.
- Meskipun teknologi ini sangat menarik, AI pada fase ini cenderung “bodoh” dan sering melakukan kesalahan atau halusinasi.

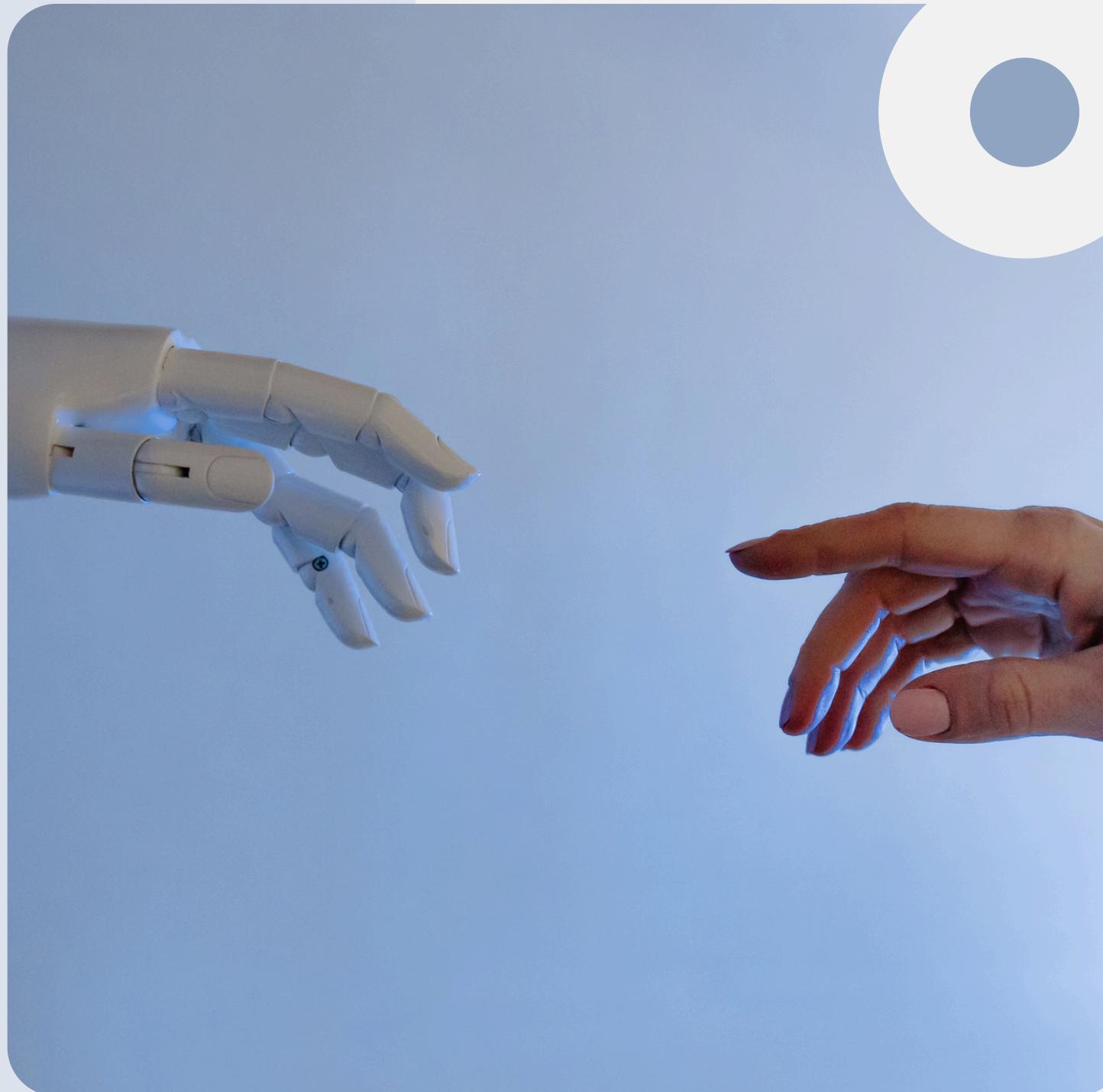
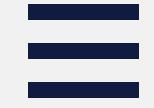
Fase 2: LLM + Context

- LLM mulai berintegrasi dengan sumber daya eksternal seperti pencarian web atau basis data internal, yang meningkatkan fungsionalitas dan nilai tambah.
- Contohnya adalah integrasi dengan aplikasi seperti Plexity (LLM + pencarian web) atau Cursor (LLM + kode sumber).

Fase 3: MCP - Menyelesaikan Masalah Integrasi

- Integrasi berbagai alat dan sumber data kini menjadi lebih rumit, menciptakan masalah besar.
- MCP hadir untuk menyelesaikan masalah ini dengan menyediakan satu protokol standar yang memungkinkan AI untuk mengakses berbagai sumber data.





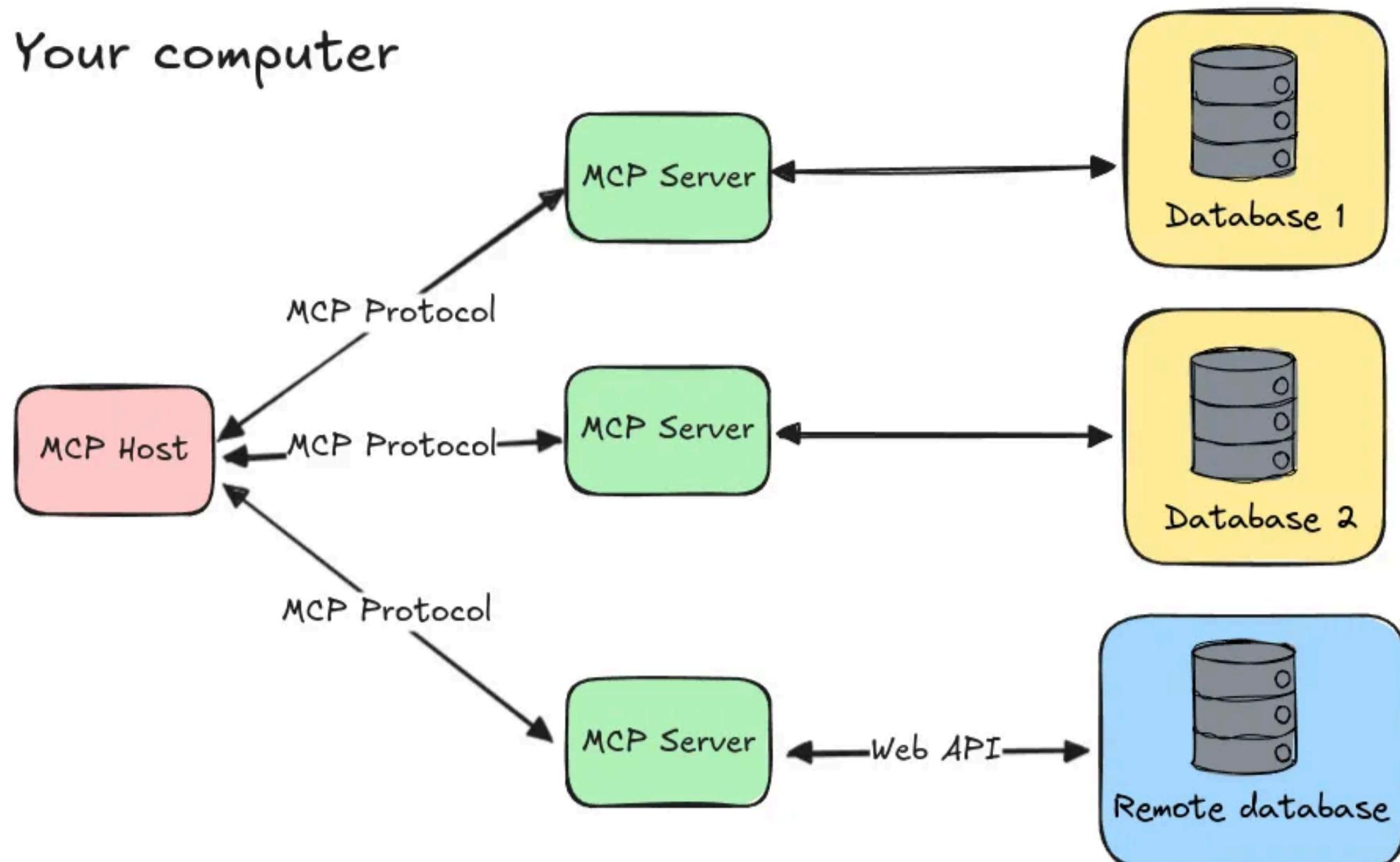
Tantangan dalam **Integrasi AI**

- Masalah utama yang dihadapi adalah kesulitan dalam menghubungkan banyak alat dan API yang berbeda dengan AI.
- Mengelola integrasi ini secara manual adalah pekerjaan yang sangat rumit dan tidak efisien. Di sinilah MCP memainkan peran penting.



Arsitektur MCP

Your computer



Arsitektur MCP terdiri dari beberapa komponen utama:

MCP Client:

Aplikasi atau agen AI yang membutuhkan data atau harus melakukan tugas tertentu. Aplikasi ini akan menginisiasi permintaan ke server MCP.

MCP Server:

Server ini berbicara dengan berbagai alat menggunakan protokol MCP dan menghubungkan berbagai sumber data lokal atau remote.

Sumber Data dan Layanan:

Sumber data terdiri dari dua jenis: lokal (misalnya file, database lokal) dan remote (misalnya API eksternal seperti Slack, GitHub, Stripe).